Software Project Management
Prof. Rajib Mall
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 33
Organization and Team Structure

Welcome to this lecture, in this lecture we will discuss about the Organization Structure and the Team Structure.

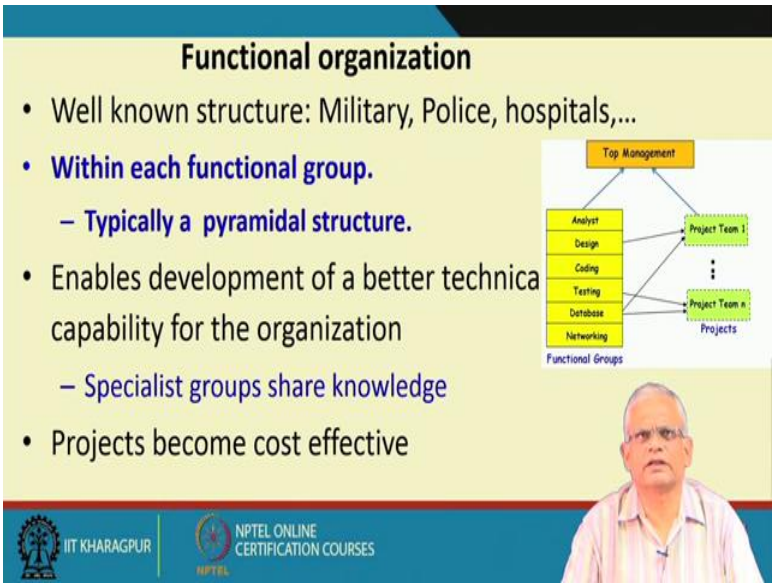
(Refer Slide Time: 00:28)

In the last lecture we had said that if you visit a Software development organization; the first thing you will notice is that the organization is structured into teams, it has many projects and different teams are carrying out the projects. The way the organization is structured into teams is called as the organization structure.

On the other hand if we look at the individual teams and how the team is structured that we call as the team structure. We had started to discuss about the organization structure that is if you walk into a software development organization, what will you notice, what are the different structures that the organization might have, how the teams are organized in the software development organization.

Broadly you will notice that any software development organization will have one of the following types of organization into teams; one is functional organization, project organization and then the matrix organization.

(Refer Slide Time: 02:02)



In the functional organization we will find functional group or departments. The projects come and go, the projects are started projects dissolve, but then what is permanent; is the functional groups are departments. There is a department for analyst, department for designers, coding, testing, database, networking and each one of these has a functional manager each department has a manager called as the functional manager.

Now, as the projects are formed a project manager is appointed; and the project manager plans the project and finds that at different points of time of the project needs different expertise. And then consults the corresponding functional manager, for example, might initially requires some analyst. Request the manager corresponding functional group manager for, the required number of analysts and as they complete their work they go back to their department and the manager would requires some designers from the design department.

Even though in a typical software development organization this kind of set up is bit rare, but then there are many places where this is a very well accepted structure. For example, military, police, hospitals, even educational institutes. In a military, the hospital is a department there are only doctors there; the doctors might be called up on for a

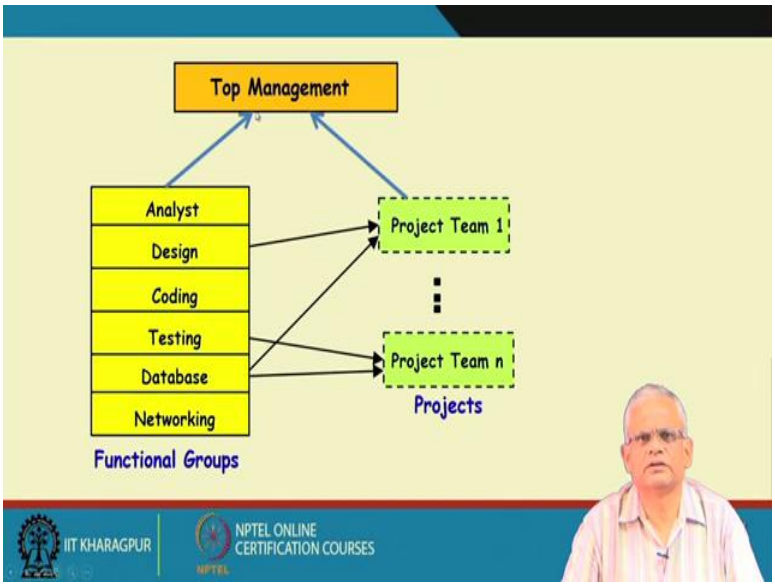
specific project. There may be administrators administration unit, there may be transportation unit, logistic units, store house, etcetera. Similar, is police and also the hospitals there are various types of specialists and these are organized into departments; for example, the surgery department, the medicine department, administration department and so on.

But if we look at the functional groups themselves; for example, let us say the design. The design department inside that if we look we will find the pyramidal structure where there is a manager is assisted by some senior designers, in turn the junior designers they report to the senior designer, the senior designer reports to the functional manager for the design unit and so on. One of the major advantage of this is that all designers they are in a department and therefore, they interact with each other, enhance their technical knowledge, educate each other and so on.

The specialist groups since the testers are all there any new tool, any new testing technology they share with each other. Another major advantage of the functional organization is that; the projects become cost effective because each time a project needs certain number of persons gets them and then they return to the respective department after they complete their work. Just imagine if they have to employ a person throughout the project duration, the designer initially he has work; but later what do he do they might ask him to do something else or he may idle.

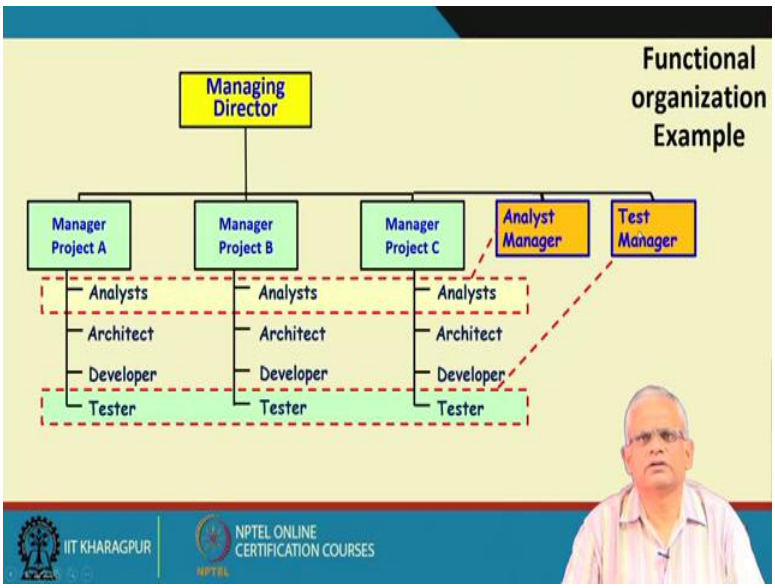
So, as far as the cost effectiveness is concerned, functional organization is very cost effective because when a specific expertize is required it is procured and retuned as soon as they complete their work.

(Refer Slide Time: 07:05)



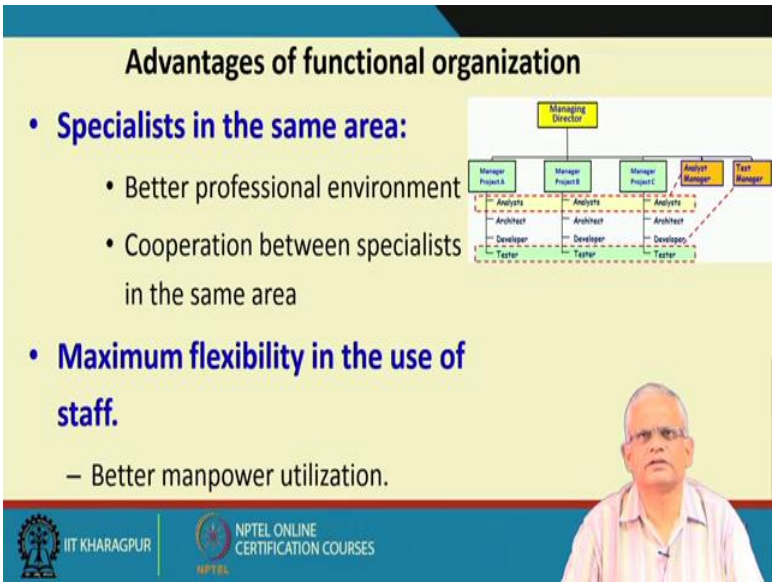
So, here the project team has taken designers and database experts who are working under project team. They would be returned as soon as they complete their work and possibly the project manager would need other expertise; and here both the functional groups and the projects, the managers here they report to the top management.

(Refer Slide Time: 07:39)



This is another view of the functional organization where each project has got some analysts, architects, developers. And they have two reporting; one is to the manager of the project and other is to the functional group manager.

(Refer Slide Time: 08:07)



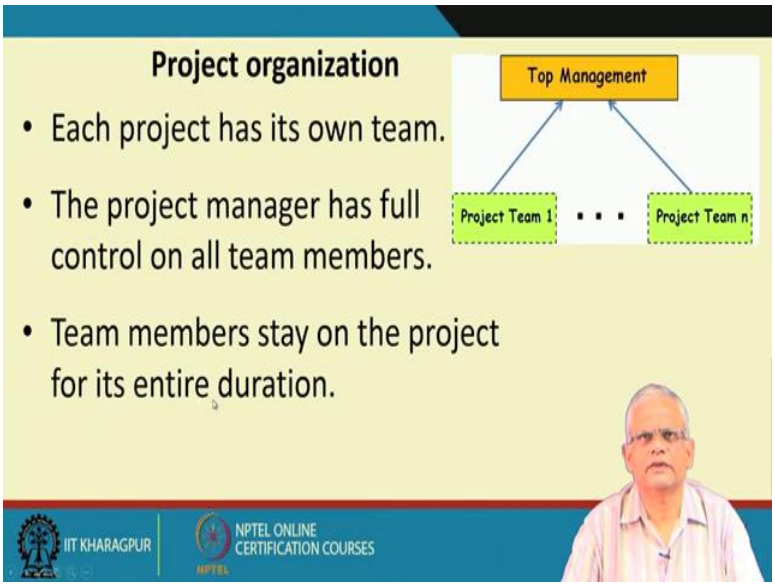
The advantages of functional organization is that better professional organization, better professional environment, cooperation between specialist; ones specialist cannot solve a problem, seeks help from the other specialist because after all they belong the same department. And possibly the biggest advantage is flexibility in use of staff, as and when manpower required get them from the department return them to the department when their work completes. The idling of the staff is minimized here, unlike we will see the other organizations.

(Refer Slide Time: 08:58)

The disadvantage of functional organization if we think of it, one is that each employee reports to two managers; one is the project manager other is the functional manager. And what if they give contradictory directives; the functional manager wants him to work in two projects and the project manager says you cannot and so on.

So, this sometimes is a stress on the workers; the project manager control decreases because he may not get exactly the persons he needs. The functional manager assigns any person that he thinks suitable. Another big problem here is that a designer keeps on doing design, a tester keeps on testing, a coder keeps on coding from one project to another project and so on. So, there is a lack of job variety.

(Refer Slide Time: 10:15)

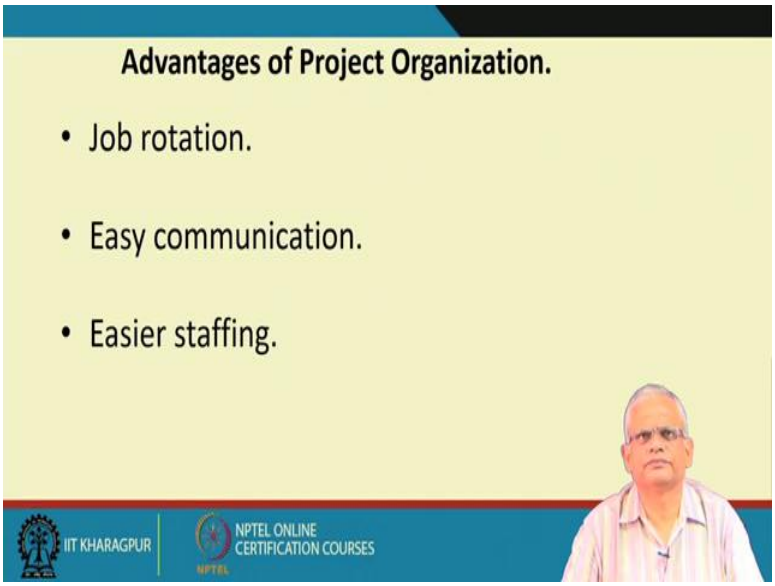


Possibly the most popular organization in software development companies is the project organization. Here when you walk into the organization you do not see the departments the department of designers, department for coders, department for testers and so on, the testing department, coding department and so on. Here what you see is the projects. The company as a whole is structured into number of projects and the workers belong to the project, they fully report to the project manager and therefore, the product managers has full control on the employees.

The team members are selected for a project and they stay on for the entire duration; but then the question is that suppose we have an analyst and then he is with some project team, initially he does requirements analysis and so on. But after that what does he do?

Ok; he also does design, also does coding testing and so on. So, here in a project organization typically each member does a variety of roles and therefore, there is a job variety here. A coder need not be coding always, he will also have option for testing, design and so on.

(Refer Slide Time: 12:11)



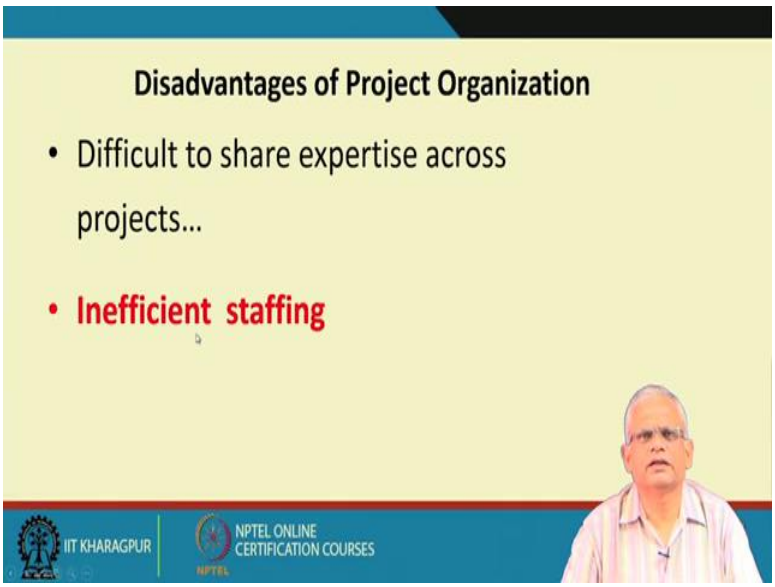
And that is a big advantage of the project organization, is the job rotation. Because there is a lot of manpower turnover in software industry. If you employ somebody as a coder he will always try to change his job and say; that may be I would like to become a designer or a analyst or a tester why should I just keep on coding all the time. That problem would arise in a functional organization, but not in a project organization because a different points of time in the same project he will be doing different activities. So, this is the big advantage in the project organization leads to job satisfaction and less manpower turnover.

The other big advantage is ease of communication. Imagine that in a functional organization the designers have designed and have gone back to their department respective department. And now the coders have done their work gone to the coding department back and the testers are working on it; and they do not understand some code some design whom do they consult, they only have some may be paper documents with them.

Communicating with them with the persons who actually designed, who actually coded may become difficult; because they might be working on an other project and when the telephone or something and try to meet and find out, they may say that we are busy on another project. So, here communications of those who do the different parts of the project become difficult.

Another big advantage of the project organization is ease of staffing. The manager has the option to select the best people for the project and they continue to stay on the project; they develop domain knowledge and the project. On the other hand in a functional organization when the project manager requests for a designer after the analysts have completed; the design manager might say that see right now we do not have designers they are all busy in respective projects and they will be given to you after 2 weeks. So, that problem does not occur in the project organization. So, this is has much easier staffing compared to the functional organization.

(Refer Slide Time: 15:13)

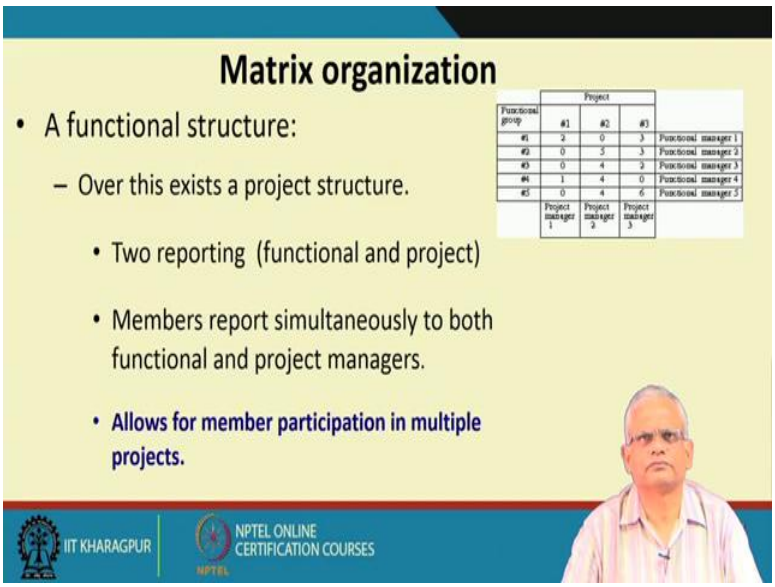


But let us look at the disadvantage of the project organization. Here there are team members continue to stay on a project, the things they learn from a project becomes difficult to share across other projects. Because the projects may run for 6 months, year and so on and they continue to work on a project and by the time they move to different project, they might have forgotten what they learnt in project.

But possibly the biggest disadvantage is inefficiency; in any project it is well known that to start with need few people in the project, in during the requirement analysis just 1 or 2 persons, during design may be 3, 4, during coding may be 10 and during testing may be 20. So, the manpower overtime in a project is not constant, it varies with time and typically given in the form of a Rayleigh distribution. But then in a project organization, the project manager appoints the project personnel at the start of the project and therefore, by definition the project organization the manpower in the project is constant.

If the project manager employees ten people, initially they have very little work, but then during design also not everybody is busy and then during coding everybody becomes stressed, there is not enough hands everybody works overtime. And during testing the situation become still worse and therefore, stress is inbuilt into the project organization; initially everybody relaxes and as the project picks up the members are under stress. Initially there is idling; later there is requirement of overworking and so on.

(Refer Slide Time: 17:45)

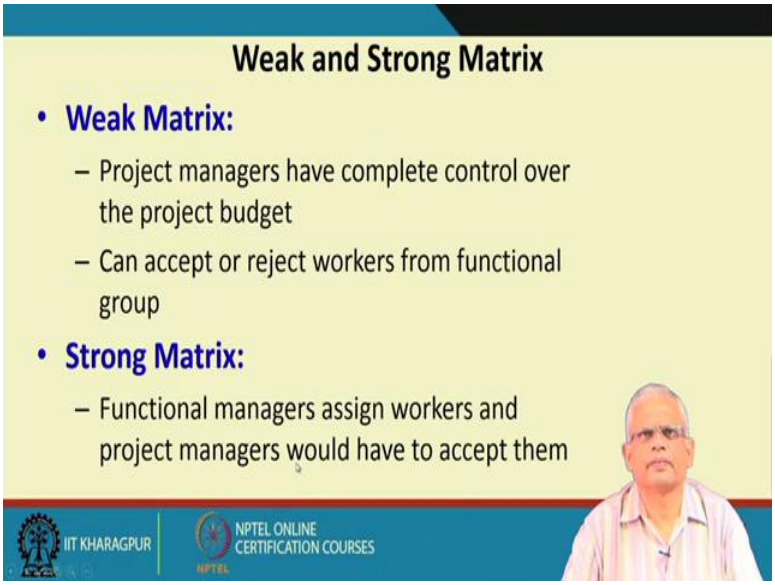


The third category of organization structure is the matrix organization. Here essentially it is a functional structure, you can find here the functional groups here, these are the departments; but then over the functional group there is a project structure you can see the projects here.

So, here again, just like a functional organization the members report to both functional group manager and the project manager. You see here that the functional group one

which may be the analyst group, there are two persons who working on project 1, no one is working on project 2 and 3 working on project 3. Here the members reports simultaneously to both functional and the project manager. But one advantage of this is that, a member in a functional group may work on two teams that is also becomes possible.

(Refer Slide Time: 19:03)



A matrix organization can be either weak or a strong matrix. In a weak matrix organization it is the project manager who has more control over the project budget, over the team members and so on. For example, he might choose a specific designer from the design department; and if the functional manager the design manager gives a different designer he may have the option to either accept or reject from the functional group.

On the other hand in a strong matrix; the functional managers have are much more powerful, they decide whom to depute for a specific project and the project managers they have to accept them.

(Refer Slide Time: 20:07)

Let us look at the advantages and disadvantages of the matrix organization. Disadvantage is just like the functional organization here each members has multiple bosses to report, the project managers have reduced control as compare to the project organization.

And in a strong matrix organization the functional manager often finds that different projects are facing problem and some projects are comfortable. So, the functional manger would like to shift manpower. The competent designers he might take from one project half way replace him; take him to another project which is facing difficulty and so on.

So, it is observed that in a strong matrix organization there is a frequent shifting of workers in a firefighting mode. Advantages just like the functional organization seems the experts belong to same departmental group, they share knowledge and also better professional growth path; because designer will become a senior designer, become a design manager and so on.

(Refer Slide Time: 21:34)

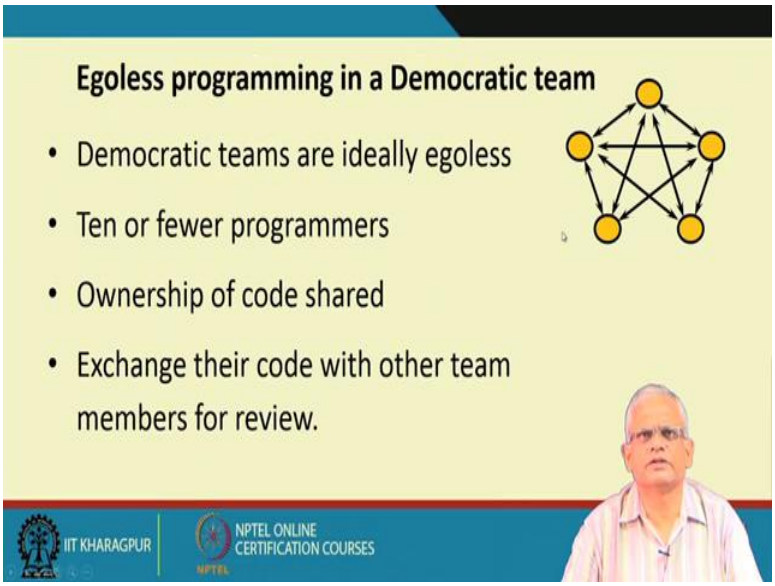
Now, let us look at the team structure. If you again work into different software development organization and look at the individual teams, you will find that the team is organized in either of the three ways; the democratic team, chief programmer team and the mixed organization. And we will see that each of these team structure has it is own advantages and disadvantages and depending on the project a specific team structure may be advantageous.

(Refer Slide Time: 22:17)

First let us look at the democratic team. These are small teams; we will see that for a large team the democratic team structure becomes very very inefficient nobody would like to have a democratic structure for a large project. Here as a name implies all technical decisions are based on consensus. Everybody can give his opinion on every issue, they talk to each other and here there is a manager who provides the administrative leadership.

But the technical part is supervised by a technical leader; but the technical leader is one among the team member who is elected by rotation. And if you look at here since everybody is consulting everybody, there is lot of time is goes for discussion. One good thing is that they can brain storm come up with good solutions, but then the other thing that it wastes lot of time and specially; if the team size is large, you can see that there are for a team size N there are N square communication path. So, it may become inefficient, too much of time may be spent in talking to all the team members and therefore, it is rarely used for large teams.

(Refer Slide Time: 24:20)

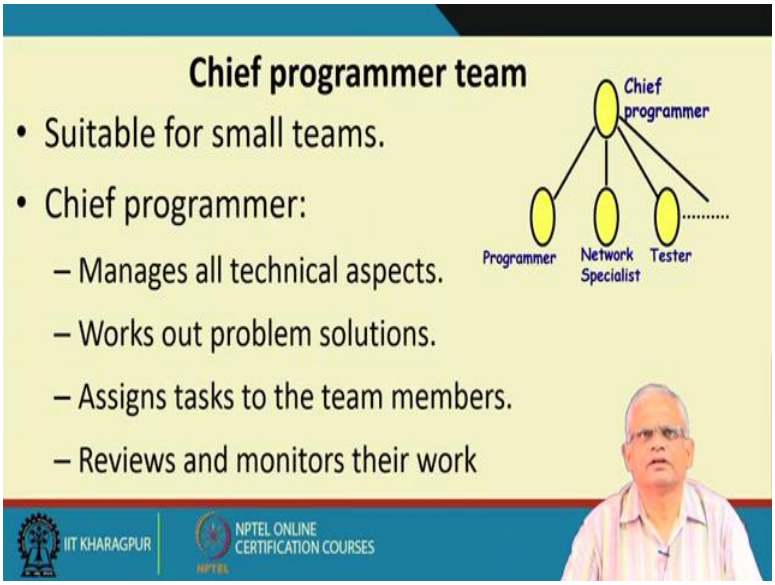


In a Democratic team ideally there can be a egoless programming environment. In a typical project every employee there, every worker has some knowledge the work that he does he is the expert. The others if you ask them that what happened to the database aspect then they may say that we do not know let the corresponding person come he will

explain. But in a democratic team these are egoless teams; that is no one owns anything there is no ownership we can't say that only he knows or one person knows.

Here the ownership is shared each team member whatever work he does is reviewed by other members. So, everybody has knowledge about every aspect of the project. This is a big advantage of the democratic team when we have egoless programming.

(Refer Slide Time: 25:41)



The third type of organization is the chief programmer team. This is again suitable for small teams where there is a chief programmer; he is responsible for all high level decisions, all technical decisions, works out the overall solution and gives small pieces of work to the different team members. And then reviews the work that is completed; monitors and reviews the work and then integrates them. We will see that this is an efficient organization for very simple problems; but then for challenging problems this may not be the right structure. We are almost at the end of the time; we will stop here and continue in the next lecture.

Thank you.